

SysTech Reporting Services

RI EPA Reports Data

FUNCTIONAL REQUIREMENTS DOCUMENT

PREPARED AND PRESENTED BY
SYSTECH INTERNATIONAL



23 JANUARY 2013
VERSION 001, REVISION 002

BUSINESS CONFIDENTIAL



DOCUMENT CONTROL

Security Classification:	Company Confidential
Document Status:	Draft v001r001
Issue Date:	2011.02.18
Author(s):	Chris Smith
Authorizer(s):	Chris Smith
Issuer:	Chris Smith
Distribution:	As per attached Distribution List
Document History:	As per attached Revision History

REVISION HISTORY

Name	Date	Change Details	Distribution	Doc Ver	Doc Rev
Luis Camacho	2011.04.27	Created version 1, revision 1 (first draft)	Chris Smith	v001	r001
Chris Smith	2013.01.24	Misc. Updates	Frank Ofiero	v001	r002

DISTRIBUTION LIST

Name	Group	Email
Chris Smith	STI	chris.smith@systech-esp.com
Bruce Kohn	STI	bruce.kohn@systech-esp.com
Frank Ofiero	STI	Frank.ofiero@systechportal.com
Nicholas Fusaro	STI	Nick.fusaro@systechportal.com
Joyce Fiore	RI DEM	joyce.fiore@dem.ri.gov

TABLE OF ABBREVIATIONS

TABLE OF CONTENTS

A. PROJECT OVERVIEW.....	5
B. THE PROCEDURE.....	6
C. APPENDIX.....	7

A. Project Overview

SysTech has created a procedure for organizing vehicle inspection records and reporting on the data as specified by the Rhode Island Department of Environmental Management (DEM).

The vehicle inspection records are organized in Test Sets. A Test Set is all the Tests for a vehicle from the initial Test until the Passed Test or until 30 days after the last failed Test.

The tests counters are reordered within a Test Set solving situations like of out of sequence test counters and/or duplicate test counters and creating a more consistent reporting on Initial Tests and Retests counts across all the EPA reports.

The test type is updated to the failed over test giving more consistent reporting on Test Type counts across all the EPA reports.

B. The Procedure

The clean up procedure is as follows:

1. SysTech selects the inspection results for all the tests in a year and we order them by VIN and KeyDateTime . The input record looks like this:

VIN	KeyDateTime	Unit	Station	VType	VYear	VFuel	Counter	Result	...
...	OBD	TSI	Transient	HDiesel	LDiesel	Func	Safety	Void	...
...	Sticker								

2. If the test is void then the eresult is 'V' and the record is skipped
3. If the Vehicle Fuel is 'D' and the Sticker is greater than 0 then the Light Diesel result is 'P'
4. If the Vehicle Fuel is 'D' then ecode is 'L'
5. If any of the tests is 'P' then eresult is 'P' otherwise if any test is 'F' then eresult is 'F' otherwise ''.
6. ECode is set to the code in that order of importance as follows :

Test	Code
OBD	O
TSI	I
Transient	N
Light Diesel	L
Heavy Diesel	H

7. When the VIN changes we reset the Tests counters
8. If the eresult is 'P' and the eresult was 'P' and it is within 30 days, the record is a duplicate and eresult is 'D' otherwise is a new Test Set.
9. The result is a new record with the new counters and Test Set numbers that looks like this:

VIN	KeyDateTime	Unit	Station	VType	VYear	VFuel	Counter	Result	...
...	OBD	TSI	Transient	HDiesel	LDiesel	Func	Safety	Void	...
...	Sticker	EResult	ECode	ECount	TestSet				

This procedure is run daily for the inspections of the current year. Previous years were already created and are not updated anymore. The new data will remain in a permanent table for reporting.

C. Appendix

```
USE [SysTest]
GO

SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE procedure [dbo].[spCreateViewTestsSummary] @Year int = 0
AS
set nocount on

declare      @VIN varchar(17);
declare      @KeyDateTime datetime;
declare      @Unit varchar(8);
declare      @Station varchar(10);
declare      @VType char(1);
declare      @VYear smallint;
declare      @VFuel char(1);
declare      @Counter int;
declare      @Result char(1);
declare      @OBD char(1);
declare      @TSI char(1);
declare      @Tran char(1);
declare      @HD char(1);
declare      @LD char(1);
declare      @Func char(1);
declare      @Saf char(1);
declare      @TestCycle char(1);
declare      @Sticker decimal(9,0);

declare      @OldVIN varchar(17);

declare      @DatePlus30 datetime;

declare      @EResult char(1);
declare      @ECode char(1);
declare      @ECount int;

declare      @TestSet int;

declare      @Reset bit;
declare      @PFound bit;
declare      @Dup bit;

if @Year = 0  set @Year = year(dateadd(d,-1,getdate()))

delete from ViewTestsSummary where year(keydatetime)=@Year;

declare curTests cursor fast_forward for
    select *
    from testsview
    where year(keydatetime)=@Year
    order by vin, keydatetime
```

```
OPEN curTests

set @Reset = 1;
set @TestSet = 0;

FETCH NEXT FROM curTests INTO @VIN, @KeyDatetime, @Unit, @Station, @VType, @VYear,
@VFuel, @Counter, @Result, @OBD, @TSI, @Tran, @HD, @LD, @Func, @Saf, @TestCycle,
@Sticker

WHILE @@FETCH_STATUS = 0
BEGIN

    if @VFuel = 'D' and @Sticker > 0 set @LD = 'P';
    if @VType = 'M' set @VType = 'T';

    set @EResult = case
        when @OBD='P' or @TSI='P' or @Tran='P' or @HD='P' or @LD='P' then 'P'
        when @OBD='F' or @TSI='F' or @Tran='F' or @HD='F' or @LD='F' then 'F'
        else ' ' end;

    set @ECode = ' ';
    set @Ecode = case when @OBD in ('F','P') then 'O' else @Ecode end;
    set @Ecode = case when @TSI in ('F','P') then 'I' else @Ecode end;
    set @Ecode = case when @Tran in ('F','P') then 'N' else @Ecode end;
    set @Ecode = case when @HD in ('F','P') then 'H' else @Ecode end;
    set @Ecode = case when @LD in ('F','P') then 'L' else @Ecode end;

    if @VFuel = 'D' set @ECode = 'L'

    if @ECode = 'O' and @VYear < 1996
        begin
            set @ECode = 'x';
            set @EResult = 'x';
        end

    if @Reset = 1
        begin
            set @ECount = 0;
            set @PFound = 0;
            set @Dup = 0;
            set @TestSet = @TestSet + 1;
        end

    if @Dup = 1
        begin
            set @EResult = 'D';
            set @ECount = 98;
        end

    set @ECount = case when @EResult = ' ' then @ECount else @ECount + 1 end;

    INSERT INTO ViewTestsSummary
        SELECT @VIN, @KeyDatetime, @Unit, @Station, @VType, @VYear, @VFuel,
        @Counter, @Result, @OBD, @TSI, @Tran, @HD, @LD, @Func, @Saf, 0,
        @Sticker, @EResult, @ECode, @ECount , @TestSet, @TestCycle

    set @Reset = 0;
    set @Dup = 0;
```



```
if @PFound = 0 and @EResult = 'P'
begin
    set @PFound = 1;
    set @DatePlus30 = dateadd(d,30,@KeyDateTime);
end

set @OldVIN = @VIN;

FETCH NEXT FROM curTests INTO @VIN, @KeyDatetimE, @Unit, @Station, @VType,
@VYear, @VFuel, @Counter, @Result, @OBD, @TSI, @Tran, @HD, @LD, @Func, @Saf,
@TestCycle, @Sticker

if @VIN <> @OldVIN set @Reset = 1;
if @PFound = 1 and @keydatetimE <= @DatePlus30 set @Dup = 1;
if @PFound = 1 and @keydatetimE > @DatePlus30 set @Reset = 1;

END

CLOSE curTests;

DEALLOCATE curTests;

select 'Done'

GO
```